WHAT IS CLAIMED:

1. A main memory simulator for simulating large computer memories, wherein said large computer memories are defined by a plurality of memory addresses, wherein each of said plurality of addresses contains data, and wherein said large computer memories are large enough to prevent simulation via the use of one-to-one memory to file addressing for all of said plurality of memory addresses, wherein said simulator comprises:

a memory cache; and

a processor operable with said memory cache, wherein said processor operates under instructions to move data contained in frequently used memory addresses of said plurality of memory addresses to said memory cache on a fast memory access basis and under instructions to move data contained in infrequently used memory addresses of said plurality of memory addresses to said memory cache on a slow memory access basis.

2. The simulator of claim 1, wherein said fast memory access comprises the utilization of a set of fast lookup tables to directly obtain a page address that has been allocated to the frequently used memory address.

3. The simulator of claim 1, wherein said slow memory access comprises the utilization of a slow lookup table to first determine if a page address has been allocated to the infrequently used memory address and to then obtain the allocated page address.

4. The simulator of claim 2, wherein said slow memory access comprises the utilization of a slow lookup table to determine if a page address has been allocated to the infrequently used memory address and the obtainment of said page address if allocated.

5. The simulator of claim 1, wherein the movement of data is achieved through a page transfer.

6. A main memory simulator for simulating large computer memories, wherein said large computer memories are defined by a plurality of memory addresses, wherein each of said plurality of addresses contains data, and wherein said large computer memories are large enough to prevent simulation via the use of one-to-one memory to file addressing for all of said plurality of memory addresses, wherein said simulator comprises:

means for storing data;

means for processing instructions, wherein said means for processing instructions is in communication with said means for storing data; and

means for providing instructions to said means for processing instructions, wherein said means for providing instructions provides the

instruction to transfer data by identifying a memory address of said data, wherein said memory address comprises at least one of said plurality of memory addresses;

wherein said means for processing instructions processes said instruction to transfer data by determining if said memory address of said data is a frequently or infrequently accessed memory address, and by transferring said data to said means for storing via a fast memory access scheme if said data is a frequently accessed memory address or by transferring said data to said means for storing via a slow memory access scheme is an infrequently addressed memory address.

7. The simulator of claim 6, wherein fast memory access scheme comprises utilizing a set of fast lookup tables.

8. The simulator of claim 7, wherein said fast lookup tables enable said means for processing to directly obtain a page address corresponding to said frequently accessed address.

9. The simulator of claim 6, wherein said slow memory access scheme comprises utilizing a slow lookup table.

10. The simulator of claim 9, wherein said slow lookup table enables said means for processing to first determine if a page address has been allocated to said infrequently addressed memory address and then obtaining the allocated page address.

11. The simulator of claim 6, wherein said transferring of data is achieved through a page transfer.

12. A method for simulating large computer memories, wherein said large computer memories are defined by a plurality of memory addresses, wherein each of said plurality of addresses contains data, and wherein said large computer memories are large enough to prevent simulation via the use of one-to-one addressing for all of said plurality of memory addresses, the method comprising:

obtaining a request for transfer of data within at least one of said plurality memory addresses, wherein said request identifies the memory address of said at least one of said plurality of memory addresses containing said data;

determining whether the requested memory address is a frequently requested memory address or whether the requested memory is an infrequently requested memory address;

if the determination reveals a frequently requested memory address, then using a fast memory access scheme to transfer the data within said frequently requested memory address; and

if the determination reveals an infrequently requested memory address, then using a slow memory access scheme to transfer the data within said infrequently requested memory address.

13. The method of claim 12, wherein said fast memory access scheme comprises utilizing a set of fast lookup tables.

14. The method of claim 13, wherein said fast lookup tables enable direct obtainment of a page address corresponding to said frequently accessed address.

15. The method of claim 12, wherein said slow memory access scheme comprises utilizing a slow lookup table.

16. The method of claim 15, wherein said slow lookup table first enables determining if a page address has been allocated to said infrequently addressed memory address and then enables obtaining the allocated page address.

17. The method of claim 12, wherein said transfer of data comprises a page transfer.

18. A memory simulation system for simulating a main memory of a computer, wherein the spaces of memory within said main memory are defined by a main memory address, the system comprising:

a plurality of files, wherein said files include a fast look-up table and a slow look-up table,

wherein said fast look-up table is operable to directly obtain a page address that has been allocated to said main memory address, and

wherein said slow look-up table is operable to first determine if a page address has been allocated to said main memory address and if a page address has been allocated to said main memory address, to obtain the allocated page address;

a cache, wherein said cache includes a buffer; and

an interface, wherein said interface receives a request for transfer of main memory between at least one of said plurality of files and said buffer, wherein said request is made through input of said main memory address to said interface, and wherein in response to said request said interface performs a page transfer between said at least one of said plurality of files and said buffer according to the page address that has been allocated to said main memory address via said fast look-up table or said slow look-up table.

19. The system of claim 18, wherein said plurality of files further include a last access look-up table, wherein said last access look-up table includes a last memory address accessed and the page address allocated to said last memory address accessed.

20. The system of claim 18, wherein upon said interface performing a page transfer, said main memory address represented by the page is used to search a breakpoint list, and wherein upon finding that a word within said page has a breakpoint set as indicated by the search of said breakpoint list, a breakpoint flag within a page descriptor of said page is set.

21. The system of claim 18, wherein both said fast look-up table and said slow look-up table are savable into a finite number of files.

22. The system of claim 21, wherein said fast look-up table and said slow look-up table are restorable from the saved files enabling a previously stopped simulation to continue.

23. The system of claim 18, wherein said fast look-up table is used to obtain a page address for memory that is accessed frequently and wherein said slow look-up table is used to obtain a page address for memory that is accessed infrequently.

24. The system of claim 23, wherein said fast look-up table is divided into a plurality of banks, each bank having a defined number of words, and wherein the frequency of access is determined by dividing said main memory address by said defined number of words to obtain a quotient value, wherein said quotient value is compared against a

predetermined value, and wherein the comparison provides an indication of whether said main memory address is frequently or infrequently accessed.

25. A memory simulation system for simulating main memory of a computer, wherein the spaces of memory within said main memory are defined by a main memory address, the system comprising:

means for receiving a data transfer request, wherein said data transfer request is defined by a main memory address;

means for determining the frequency of use of said main memory address;

means for obtaining a page address corresponding to said main memory address, wherein said means for obtaining includes:

means for obtaining a page address corresponding to said main memory address when said main memory address has been determined to be frequently used; and

means for obtaining a page address corresponding to said main memory address when said main memory address has been determined to be infrequently used;

means for transferring data to a memory location, wherein said transferring of data comprises a page transfer in accordance with the obtained page address.

26. The system of claim 25, wherein said means for obtaining the page address of the frequently used main memory address comprises a set of fast lookup tables.

27. The system of claim 26, wherein said fast lookup tables enable said means for obtaining to directly obtain the page address of the frequently used main memory address.

28. The system of claim 25, wherein said means for obtaining the page address of the infrequently used main memory address comprises a slow lookup table.

29. The system of claim 28, wherein said slow lookup table enables said means for obtaining to first determine if a page address has been allocated to said main memory address and then to obtain that allocated pages address.

30. A method of memory transfer for use in simulating a main memory, wherein the spaces of memory within said main memory are defined by a main memory address, the method comprising:

obtaining a main memory address, wherein said main memory address indicates a request for a main memory transfer between a file and a cache buffer;

determining if said main memory address comprises memory that is accessed frequently or infrequently;

if said main memory address comprises memory that is accessed frequently, directly obtaining a page address that has been allocated to said main memory address through use of a first look-up table;

if said main memory address comprises memory that is accessed infrequently, first determining whether a page address has been allocated to said main memory address, then obtaining the page address that has been allocated to said main memory address;

transferring the requested main memory between said file and said cache buffer via a page transfer that utilizes the page address allocated to said main memory address.

31. The method of claim 30, further comprising the step of saving the last main memory address accessed and the corresponding page address.

32. The method of claim 30, further comprising the step of searching a breakpoint list, wherein upon discovering that a word within the transferred page has a breakpoint set as indicated by the search of said breakpoint list, setting a breakpoint flag within a page descriptor of the transferred page.

33. The method of claim 30, further comprising the step of saving said first look-up table and said second look-up table into a number of finite files.

34. The method of claim 33, further comprising the step of restoring said first look-up table and said second look-up table to continue a previously stopped simulation.

35. The method of claim 30, wherein said first look-up table is divided into a plurality of banks, each bank having a defined number of words, and wherein said step of determining is achieved by dividing said main memory address by said defined number of words to obtain a quotient value, then comparing said quotient value against a predetermined value, wherein the comparison provides an indication of whether said main memory address is frequently or infrequently addressed.